

DREAMY TALES



Jaskaran Ram – Gabriele Shu

10735884 -11016752

Design and Implementation of Mobile app

Dreamy Tales

This is a sophisticated mobile application tailored for children, offering an immersive storytelling experience. The application enables users to profile the protagonists of stories according to their preferences. Through the integration of artificial intelligence, it dynamically generates customized narratives based on parental settings. The choices provided encompass a broad spectrum of themes, ranging from classical to avant-garde, offering diverse lessons and extensive options for character customization.



PERSONALIZE YOUR CHARACTERS



16:34 B. 100% 90%

← Add a new main character

Name
Luke

Gender
Male

Age: 3

Choose up to 5 random avatars >

Save

16:34 B. 100% 90%

☰ Dreamy Tales

Main Characters:

Luke Sophie + Add

Second Characters:

Mom + Add

★ Let's start the magic

16:34 B. 100% 90%

← Edit Second Character

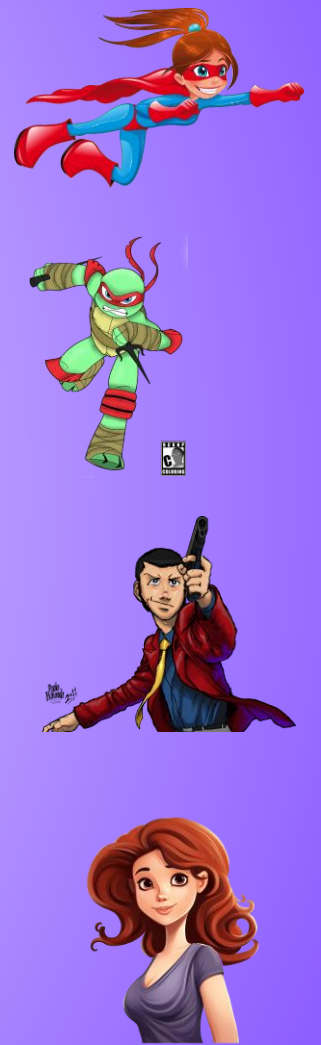
Name
Mom

Gender
Female

Role
Parent

Choose up to 5 random avatars >

Save



CHOOSE THE THEMES



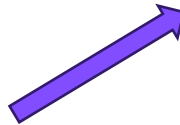
APP FLOW



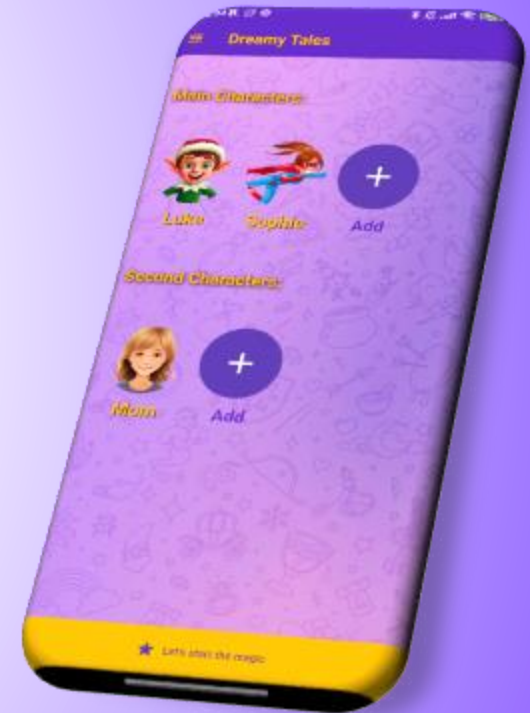
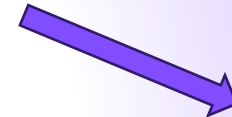
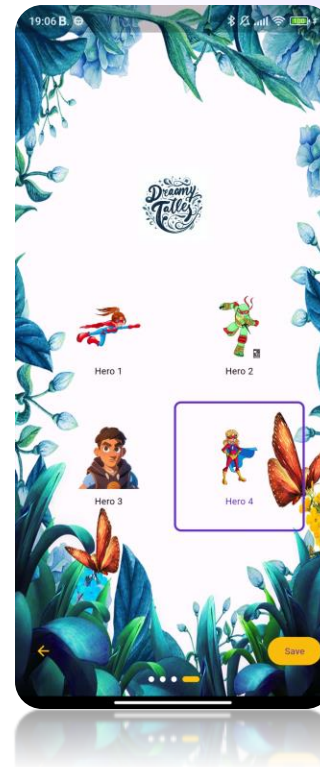
**Tutorial
(First Run)**



Login/Register

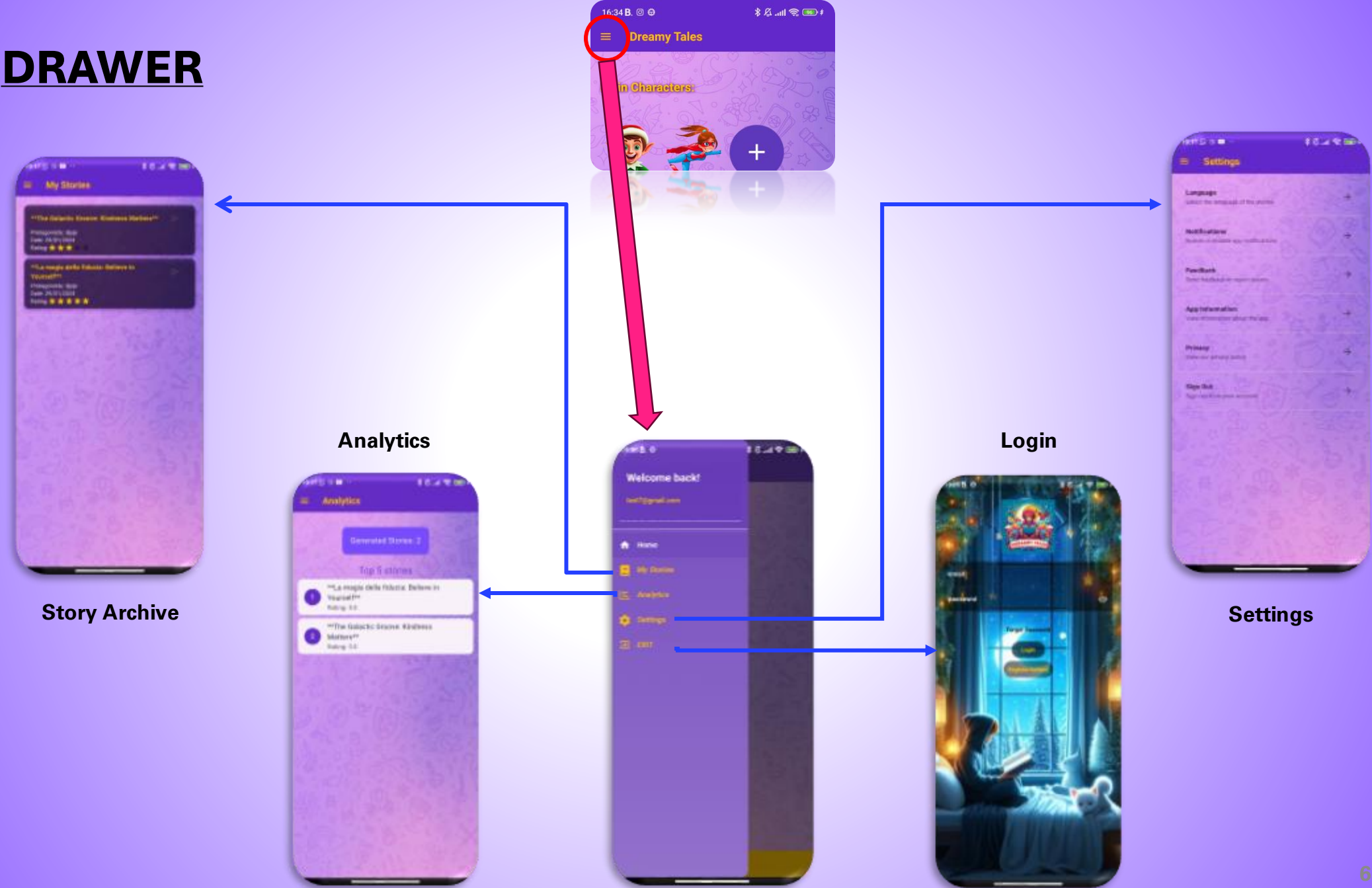


Register



**Home Page
(directly here if you are
already logged in)**

DRAWER

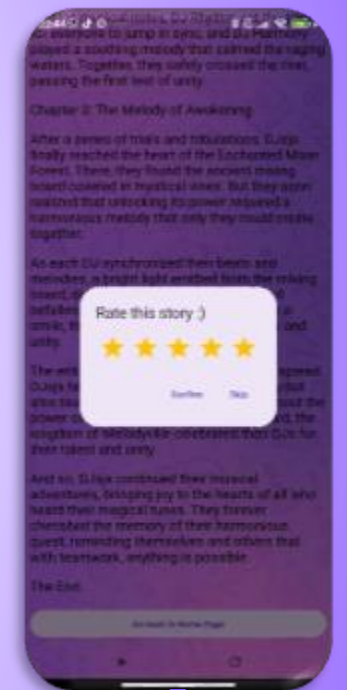
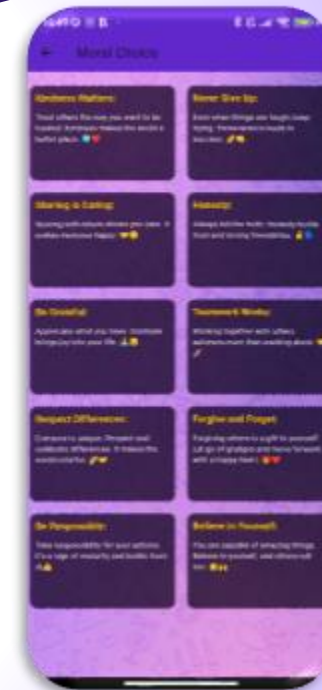


CHARACTER EDITING

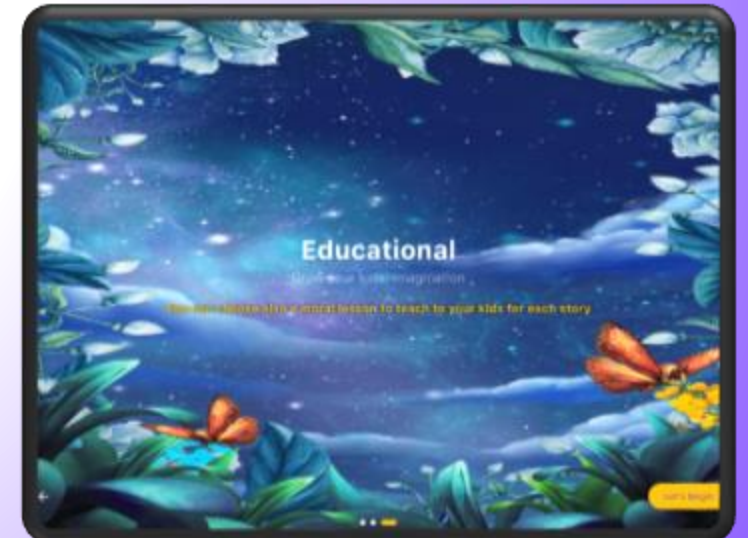


The sequence of screens illustrates the user journey:

- Character Selection:** The user selects main characters (Luke, Sophie) and second characters (Mom). A red box highlights the "Let's start the magic" button at the bottom.
- Genre Selection:** The user chooses between "Fantasy" and "Classic" genres.
- Story Selection:** The user selects a specific story from a list including "Marvel", "Sleeping Beauty", and "Cinderella".
- Story Choices:** The user interacts with a series of choices that branch the story, such as "Children's Math", "Never Give Up", "Sharing is Caring", "Honesty", "Be Careful", "Teamwork & Work", "Respect Differences", "Forgive and Forget", "Be Responsible", and "Believe in Yourself".
- Story Completion:** The user rates the story (5 stars) and reaches the end screen, which says "The End" and "Go back to Home Page".

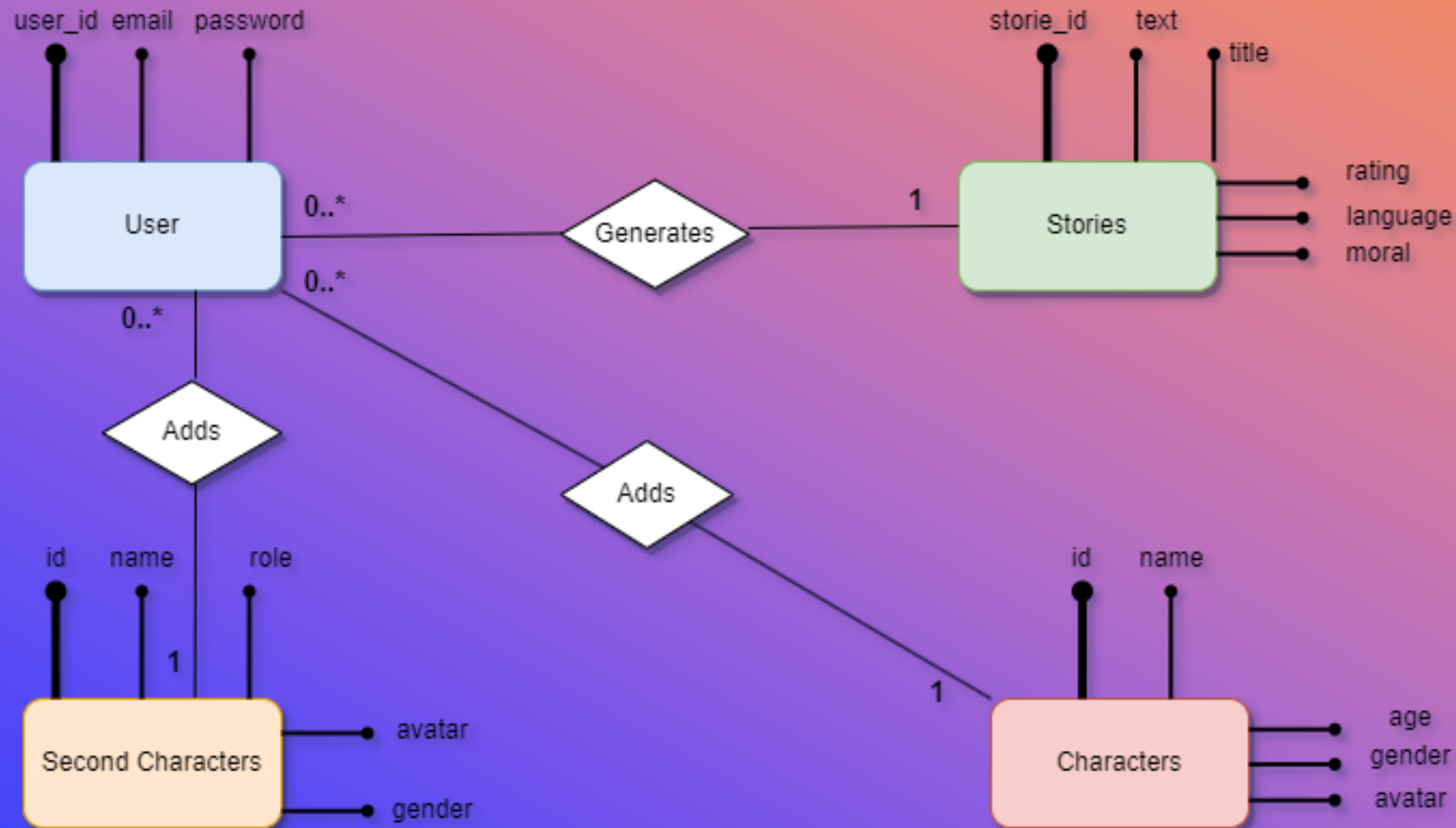


MULTIPLATFORM AND MULTI-DEVICE APPLICATION

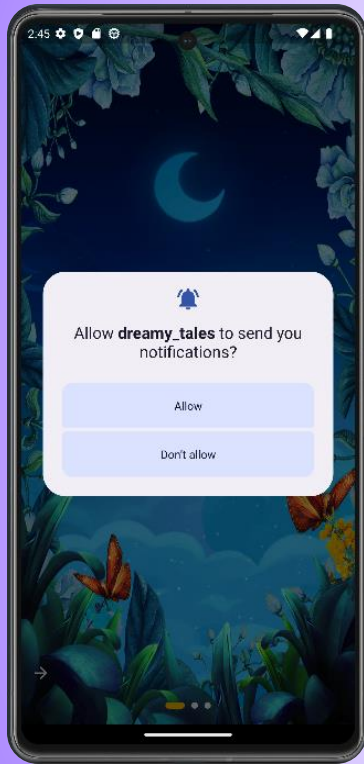


DATABASE DIAGRAM

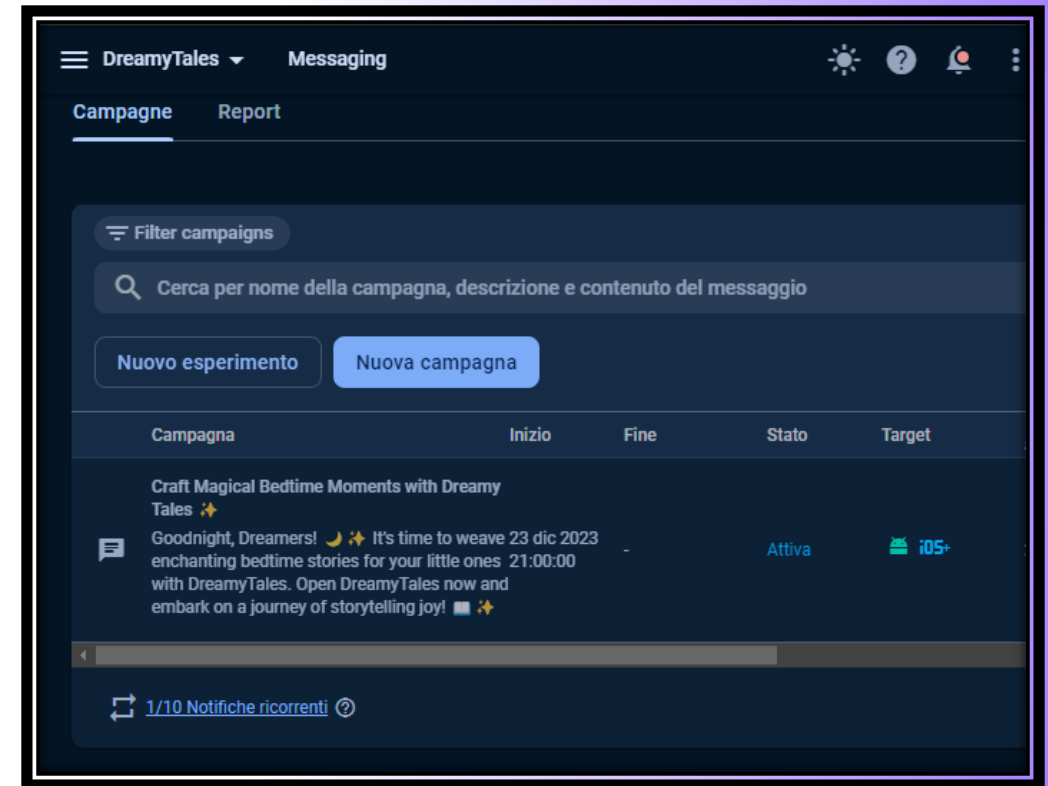
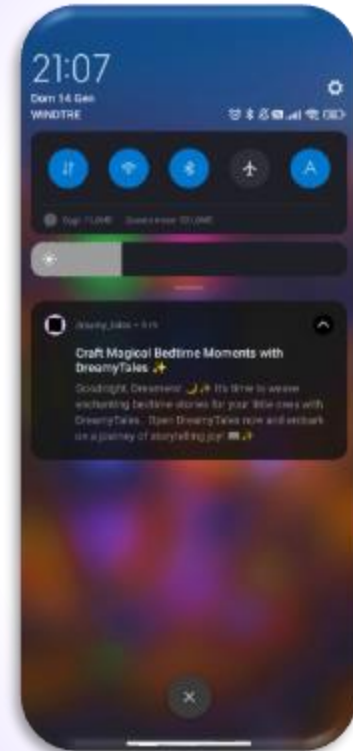
DIMA 2023/24



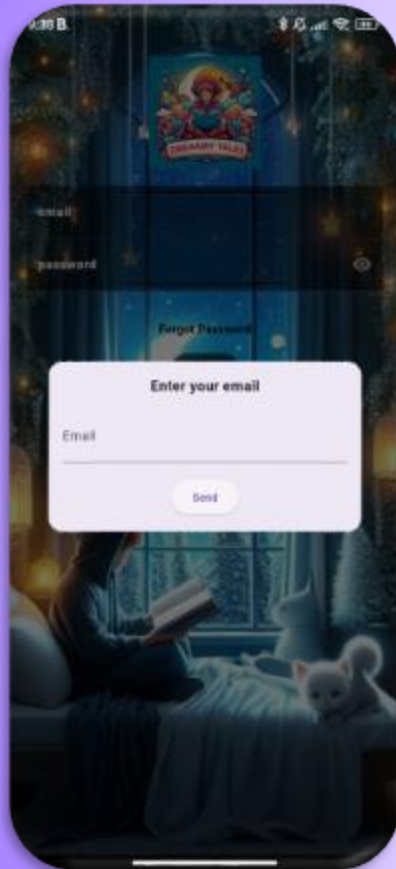
Features



Notifications



Features



Forgot Password



Share your Story



Voice Over

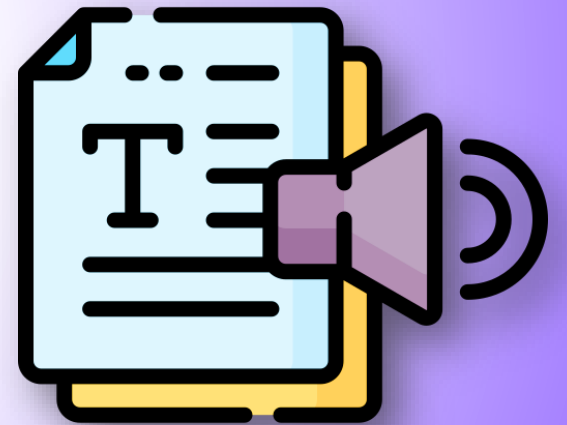


Feedback

EXTERNAL API



Firebase



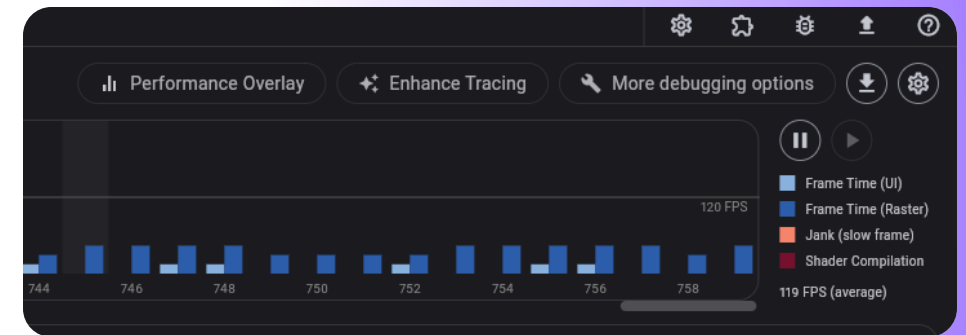
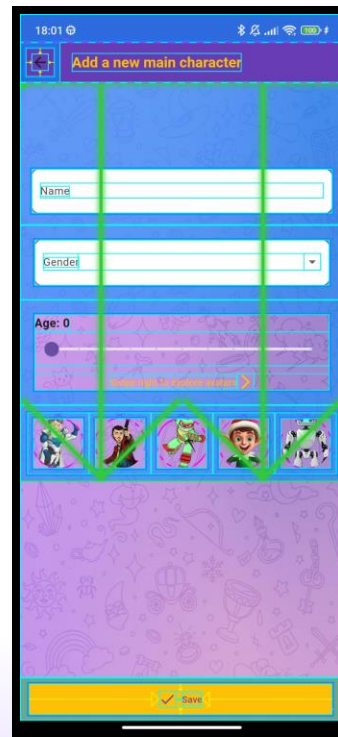
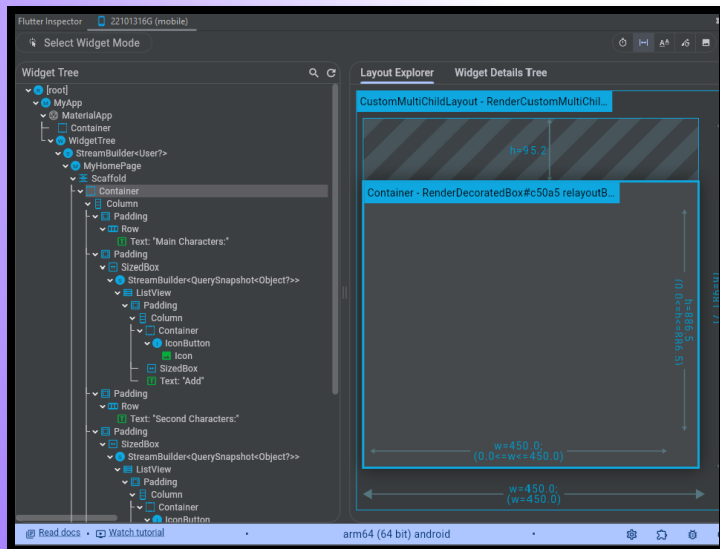
STABLE APPLICATION

Flutter DevTools :

- Performance : Find peak of low frame due some GUI compilation
- CPU profiler : sample critical actions and evaluate if it is needed some optimization

Flutter Inspector :

- Analyze Widget Tree
- Layout Explorer to fix positional issues

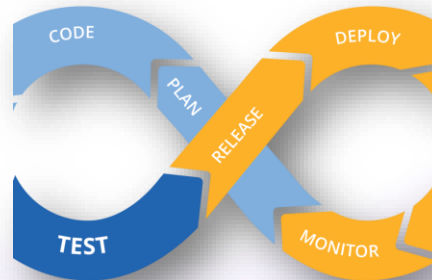


The screenshot shows the 'CPU Profiler' in Flutter DevTools. It displays a table of CPU usage data for various methods. The table has columns for 'Bottom Up', 'Call Tree', 'Method Table', and 'CPU Flame Chart'. The 'Method Table' column shows the method names and their corresponding CPU usage percentages.

Bottom Up	Call Tree	Method Table	CPU Flame Chart
Total Time	Self Time	Method	
172.84 ms (2.40%)	172.84 ms (2.40%)	> _postEvent - (dart:developer-patch/developer.dart:64)	
127.36 ms (1.77%)	127.36 ms (1.77%)	> FfiTrampoline___drawImageRect\$Method\$FfiNative\$Ptr - (dart:ffi:1)	
118.26 ms (1.64%)	118.26 ms (1.64%)	> FfiTrampoline___shift\$Method\$FfiNative\$Ptr - (dart:ffi:1)	
109.16 ms (1.52%)	109.16 ms (1.52%)	> FfiTrampoline___addRect\$Method\$FfiNative\$Ptr - (dart:ffi:1)	
100.07 ms (1.39%)	100.07 ms (1.39%)	> FfiTrampoline___paint\$Method\$FfiNative\$Ptr - (dart:ffi:1)	
100.07 ms (1.39%)	100.07 ms (1.39%)	> FfiTrampoline___drawRect\$Method\$FfiNative\$Ptr - (dart:ffi:1)	
95.52 ms (1.33%)	95.52 ms (1.33%)	> FfiTrampoline___dispose\$Method\$FfiNative\$Ptr - (dart:ffi:1)	
90.97 ms (1.26%)	90.97 ms (1.26%)	> FfiTrampoline___pushOffset\$Method\$FfiNative\$Ptr - (dart:ffi:1)	
86.42 ms (1.20%)	86.42 ms (1.20%)	> FfiTrampoline___dispose\$Method\$FfiNative\$Ptr - (dart:ffi:1)	

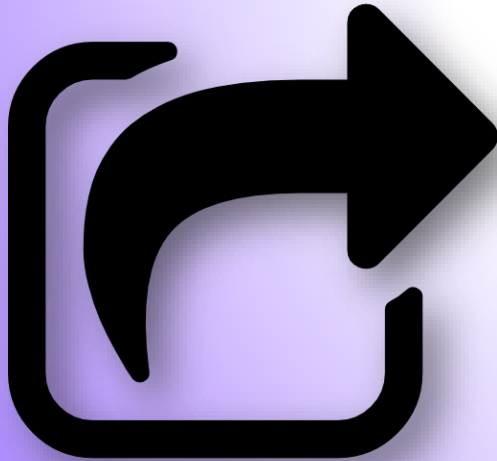
Testing

- **Unit testing:** it ensures that the end-user (customers) can achieve the goals set in the business requirements, which determines whether the software is acceptable for delivery or not.
- **Widget testing:** the goal of a widget test is to verify that the widget's UI looks and interacts as expected.
- **Integration testing:** it ensures that an entire, integrated system meets a set of requirements. It is performed in an integrated hardware and software environment to ensure that the entire system functions properly.



Future Extension

- Share story between app users
- Recommender System about Stories
- Improve Analytics with more details



THANK YOU NOW LET'S DEMO

DREAMY TALES



DIMA 2023/24

+

○

